

# RTES/CMS Work in Progress

Reported by  
Michael J. Haney, UIUC

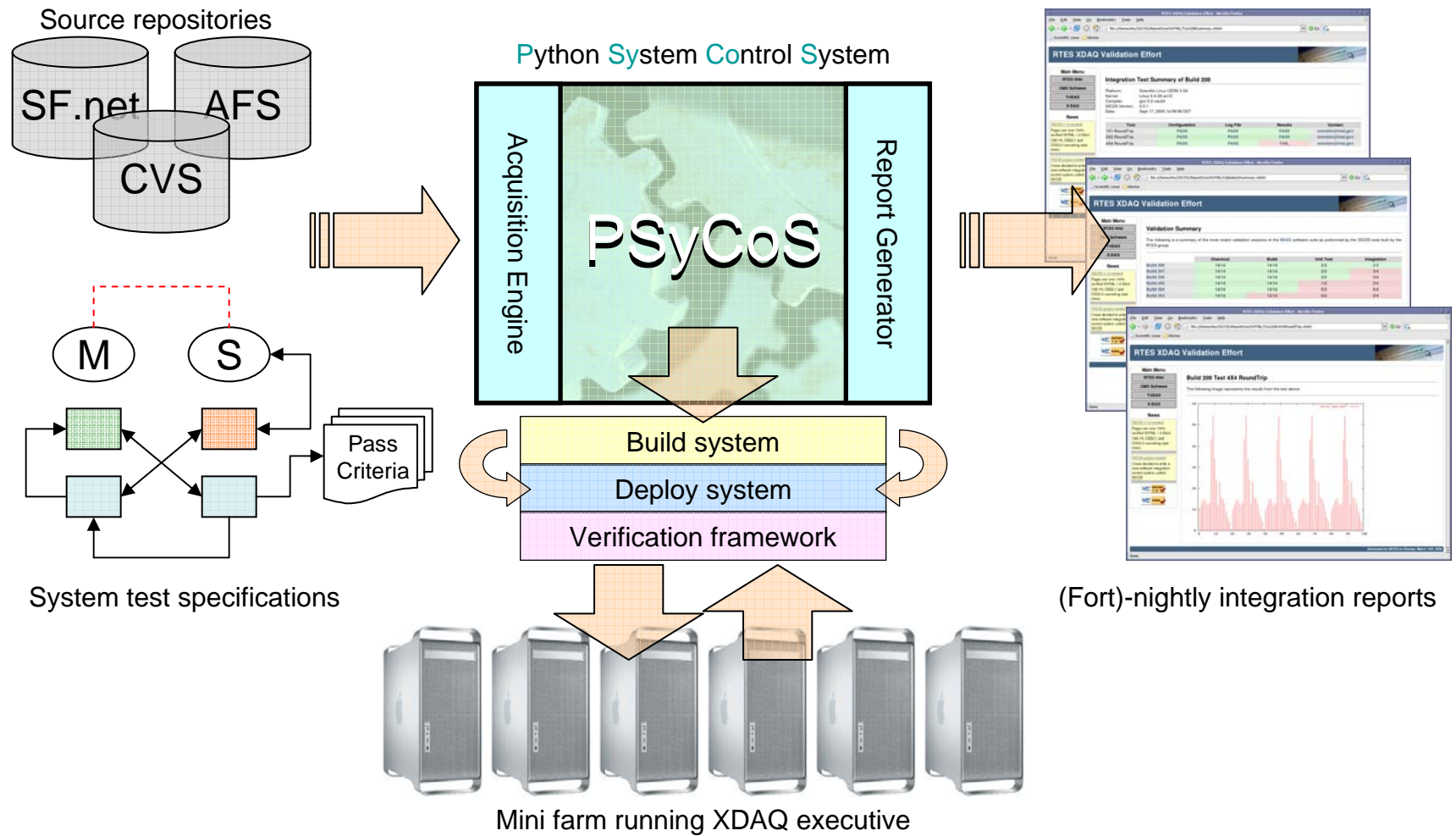
# Overview

- System Level Testing
- Web Service Testing
- Unit Testing
- Other - VLAs

# System Level Testing

- Ted Bapty, Sandeep Neema, Steve Nordstrom, Turker Kreskinpala, Abhishek Dubey – Vanderbilt
- Work in progress
- Problem: Software being developed ‘en mass’ by various teams with little integration or verification of component interaction
  - Rapidly changing interfaces
  - Releases being made very frequently
  - Non-uniform source repositories (CVS, Sourceforge, AFS, etc.)
- Solution: Automated system level testing – XDAQ and apps
  - Short term: automated software integration and verification framework (PSyCoS) which can acquire, build, deploy, and verify software at the systems integration level
  - Long term: creation of domain specific modeling language(s) to capture system structure, application configuration, hardware resources, test parameters, and verification criteria
  - Model-based code system test generators leverage the verification framework to allow fully automated system tests and report generation

# System Level Testing

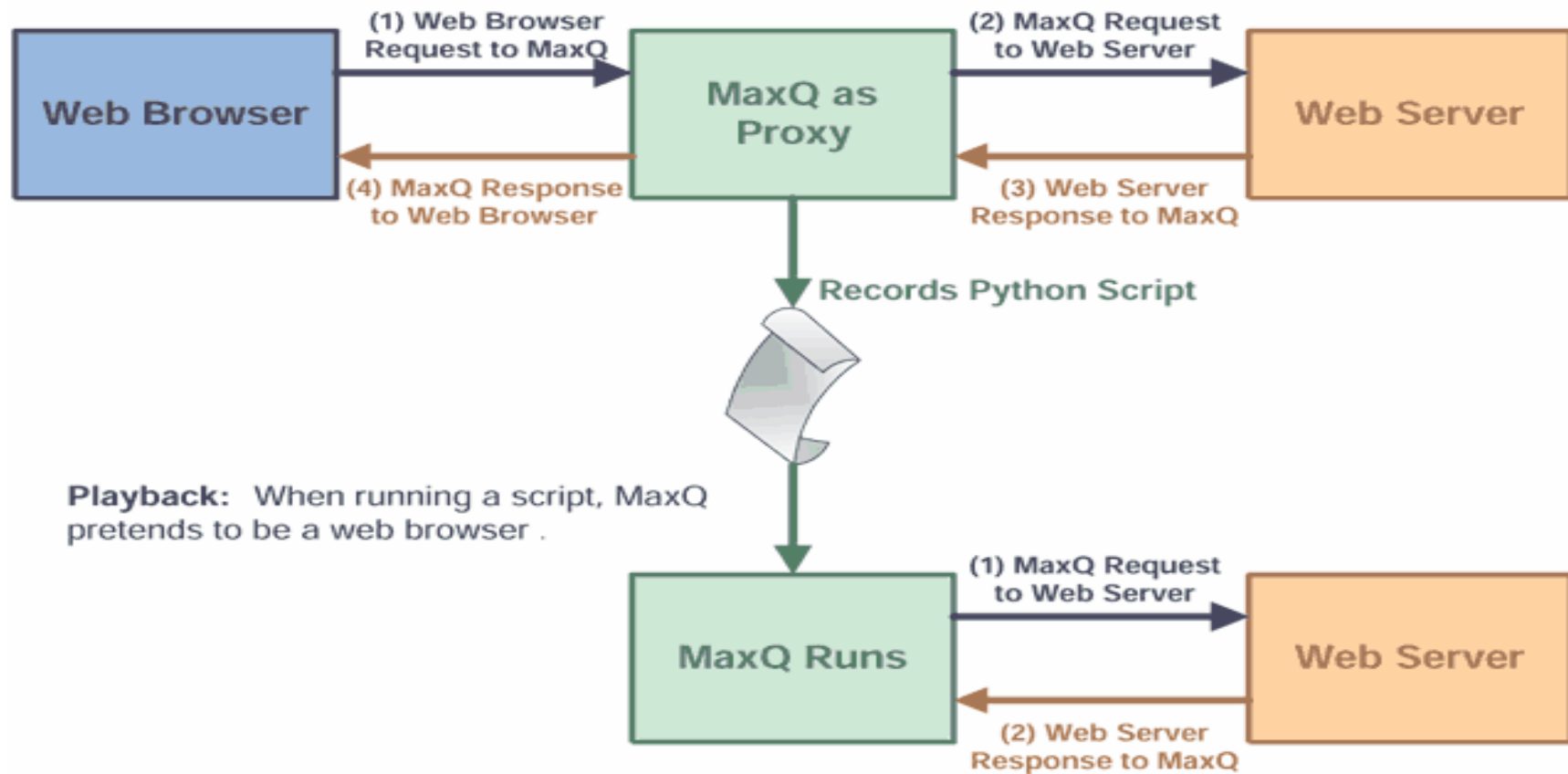


# Web Service Testing

- Mike Haney - UIUC
- Work in progress
- To test RunControl in a repeatable, automated manner
  - Using MaxQ

# MaxQ

**Recording:** MaxQ works as a proxy server for your web browser. This means that your browser's requests, and the responses from the server, go via MaxQ.



# Jython example

- `# Generated by MaxQ [com.bitmechanic.maxq.generator.CompactGenerator]`
- `from CompactTest import CompactTest`
- `class run_simple(CompactTest):`
- `# Recorded test actions.`
- `def runTest(self):`
- `self.msg('Test started')`
- `self.get('http://129.59.129.193:8080/rcms/', None, 302)`
- `self.get('http://129.59.129.193:8080/rcms/gui/servlet/RunGroupChooserServlet', None, 200)`
- `self.get('http://129.59.129.193:8080/rcms/gui/servlet/RunGroupChooserServlet', [`
- `('j_username', 'rcms'),`
- `('j_password', 'rcms2'),`
- `], 200)`
- `self.get('http://129.59.129.193:8080/rcms/gui/servlet/RunGroupChooserServlet', [`
- `('ACTION', 'ENABLED'),`
- `('directoryID', '2'),`
- `('parentID', '1'),`
- `], 200)`
- `self.get('http://129.59.129.193:8080/rcms/gui/servlet/FMPilotServlet', [`
- `('directoryID', '3'),`
- `], 200)`
- `self.post('http://129.59.129.193:8080/rcms/gui/servlet/FMPilotServlet', [`
- `('ACTION', 'Create'),`
- `('COMMAND', ''),`
- `('LAST_NOTIFICATION_STATE', ''),`
- `('groupID', '1~1'),`
- `], 200)`
- `self.post('http://129.59.129.193:8080/rcms/gui/servlet/FMPilotServlet', [`
- `('ACTION', 'Refresh'),`
- `('COMMAND', 'GoToHello'),`
- `('LAST_NOTIFICATION_STATE', 'Hello'),`
- `('groupID', '1~1'),`
- `], 200)`

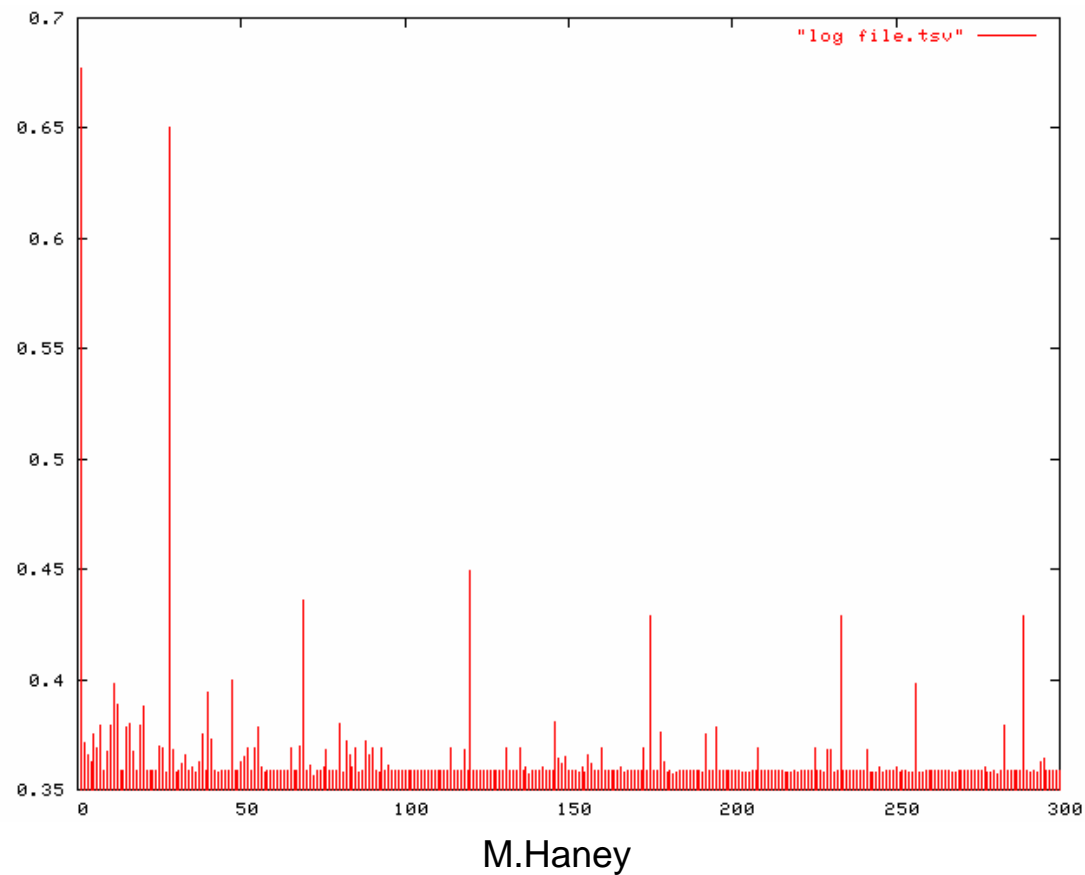
# Web Service Testing – Status

- Replicated RunControl environment on one of the Vanderbilt (RTES) machines
  - Created self-contained Makefile (wget's and cvs co's) for easy re-creation
- Recorded simple web excursion
  - “Hello” finite state machine
- Tailored Jython to repeat, and measure execution time
- Next: real testing of real state machines



# (meaningless) Results

- demonstration of concept –
  - 300 (single) state machine creations + cycles (in seconds)



# Unit Test

- Mike Haney - UIUC
- Plan
  - Review (rewrite?) and generalize xdata package (which already uses CppUnit) to provide template/guidance for others
  - Incorporate into PSyCoS for fortnightly build testing
  - Document and present, then consult/collaborate
    - Not practical for one person to write all tests for everyone...

# Very Lightweight Agents (RTES)

- Jae Oh, Derek Messie - Syracuse
- Ideas still being developed...
- For fault detection
  - Could assist the Sentinel process
    - To recognize more subtle fault conditions
  - Could be incorporated in the Job Control process
    - Which “owns” all application tasks
  - Could interact with the Problem Solver to mitigate fault impact
- Starting with our RunControl makefile/replica